

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-171522

(43)Date of publication of application : 14.06.2002

(51)Int.Cl.

H04N 7/24

(21)Application number : 2000-316686

(71)Applicant : VICTOR CO OF JAPAN LTD

(22)Date of filing : 17.10.2000

(72)Inventor : ASAMI TOMOHIRO
ANDO ICHIRO

(30)Priority

Priority number : 2000211616

Priority date : 12.07.2000

Priority country : JP

2000255172

25.08.2000

JP

2000269781

06.09.2000

JP

2000277796

13.09.2000

JP

2000288180

22.09.2000

JP

(54) DIVISION METHOD, TRANSMISSION METHOD AND INTEGRATING METHOD FOR STRUCTURAL META DATA

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a division method, a transmission method and an integrating method for structural meta data where the structural meta data are divided into files in a re-integration possible way and controlling the frequency of the transmission of data by each of the division files can enhance the transmission efficiency.

SOLUTION: It is required to imbed information as to connecting points into an MPEG-7 description in order to re-integrating divided MPEG-7 description into the one MPEG-7 description. Then reference to a DS (Description Scheme) having been attached to the description before the division under a new division route and the new division route is referenced to the DS of the divided description. In Figure 1, a DS2 of a description source is referenced to a DS3 separated from the description and the DS3 is referenced to the DS2 of the division source.

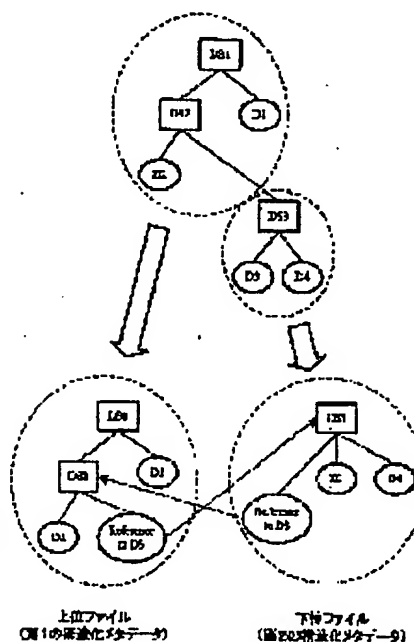


図1

LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-171522

(P2002-171522A)

(43) 公開日 平成14年6月14日 (2002.6.14)

(51) Int.Cl.⁷

H04N 7/24

識別記号

F I

H04N 7/13

テームト* (参考)

Z 5 C 0 5 9

審査請求 未請求 請求項の数33 O L (全 37 頁)

(21) 出願番号 特願2000-316686(P2000-316686)

(22) 出願日 平成12年10月17日 (2000.10.17)

(31) 優先権主張番号 特願2000-211616(P2000-211616)

(32) 優先日 平成12年7月12日 (2000.7.12)

(33) 優先権主張国 日本 (J P)

(31) 優先権主張番号 特願2000-255172(P2000-255172)

(32) 優先日 平成12年8月25日 (2000.8.25)

(33) 優先権主張国 日本 (J P)

(31) 優先権主張番号 特願2000-269781(P2000-269781)

(32) 優先日 平成12年9月6日 (2000.9.6)

(33) 優先権主張国 日本 (J P)

(71) 出願人 000004329

日本ビクター株式会社

神奈川県横浜市神奈川区守屋町3丁目12番地

(72) 発明者 浅見 知弘

神奈川県横浜市神奈川区守屋町3丁目12番地 日本ビクター株式会社内

(72) 発明者 安藤 一郎

神奈川県横浜市神奈川区守屋町3丁目12番地 日本ビクター株式会社内

Fターム(参考) 5C059 KK15 MA00 RB02 SS02 UA02

最終頁に続く

(54) 【発明の名称】 構造化メタデータの分割方法、伝送方法、及び統合方法

(57) 【要約】

【課題】 構造化メタデータを再統合可能なように複数のファイルに分割し、その伝送において、分割されたファイルごとに伝送の頻度を制御することにより伝送効率を高めることを可能とする構造化メタデータの分割方法、伝送方法、及び統合方法を提供する

【解決手段】 分割した複数のMPEG-7記述を再統合して1つのMPEG-7記述を得るためには、分割する際に接続点についての情報を記述の中に埋め込む必要がある。そこで、分離した新たなルートの下に分離前に付いていたDSへの参照を追加し、また分離された側のDSの下にも分離した新たなルートへの参照を付ける。図1では分離したDS3の下に分離元であるDS2への参照を付けるとともに、分離元のDS2の下にはDS3への参照を付ける。

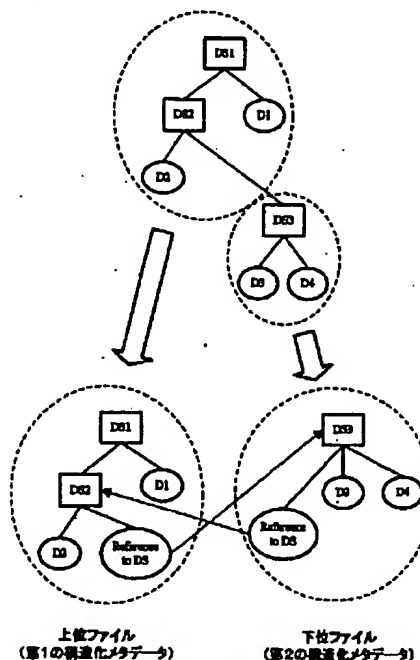


図1

【特許請求の範囲】

【請求項1】コンテンツの内容を記述するために、記述子と記述子群をノードとして木構造で構成した構造化メタデータを、第1の構造化メタデータとその第1の構造化メタデータに対して下位の関係となる第2の構造化メタデータとに分割する方法であって、

第1の構造化メタデータと第2の構造化メタデータとの双方に、または、前記第1の構造化メタデータの方に、この2つの構造化メタデータの接続点を示す情報である参照記述子を付加して分割することを特徴とする構造化メタデータの分割方法。

【請求項2】請求項1記載の構造化メタデータの分割方法において、

前記第2の構造化メタデータに前記第1の構造化メタデータに含まれる記述子群に対応するダミーの記述子群を付加して分割し、

前記第1の構造化メタデータに付加する参照記述子は、前記ダミーの記述子群を接続点とするものであり、

前記第2の構造化メタデータに参照記述子を付加する場合のその参照記述子は、前記ダミーの記述子群の要素として付加され、その付加される参照記述子は前記ダミーの記述子群に対応する前記第1の構造化メタデータに含まれる記述子群を接続点とするものである、ことを特徴とする構造化メタデータの分割方法。

【請求項3】請求項1または2記載の構造化メタデータの分割方法において、

前記第1の構造化メタデータと前記第2の構造化メタデータとで、同一の記述子及び同一の記述子群の内の少なくとも一方を重複させて分割することを特徴とする構造化メタデータの分割方法。

【請求項4】請求項3記載の構造化メタデータの分割方法において、

前記第1の構造化メタデータと前記第2の構造化メタデータとに重複させる記述子及び／または記述子群は、一方の構造化メタデータ側において、他方の構造化メタデータ側よりも記述内容量が少ないことを特徴とする構造化メタデータの分割方法。

【請求項5】請求項4記載の構造化メタデータの分割方法において、

前記第1の構造化メタデータと前記第2の構造化メタデータとに重複させる記述子及び／または記述子群は、前記第1の構造化メタデータ側において、前記第2の構造化メタデータ側よりも記述内容量が少ないことを特徴とする構造化メタデータの分割方法。

【請求項6】請求項1～5のいずれか一つに記載の構造化メタデータの分割方法において、

前記第1の構造化メタデータに付加する参照記述子は、前記第2の構造化メタデータの存在場所を示す参照先情報を有するものであり、前記第1の構造化メタデータは前記参照先情報が異なる複数の参照記述子が付加され

たものであることを特徴とする構造化メタデータの分割方法。

【請求項7】請求項1～6のいずれか一つに記載の構造化メタデータの分割方法により分割した構造化メタデータの伝送方法であって、

前記第1の構造化メタデータの繰り返し伝送頻度を、前記第2の構造化メタデータの繰り返し伝送頻度よりも高くしたことを特徴とする構造化メタデータの伝送方法。

【請求項8】請求項1～6のいずれか一つに記載の構造化メタデータの分割方法により分割した構造化メタデータの伝送方法であって、

前記第1の構造化メタデータの伝送路を、前記第2の構造化メタデータの伝送路よりも伝送品質の高い伝送路としたことを特徴とする構造化メタデータの伝送方法。

【請求項9】請求項1記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、

前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【請求項10】請求項2記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、

前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【請求項11】請求項3記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、

前記第1の構造化メタデータと前記第2の構造化メタデータとの双方に重複した記述子及び／または記述子群を、前記第1の構造化メタデータまたは前記第2の構造化メタデータから消去し、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【請求項12】請求項4記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、

前記第1の構造化メタデータと前記第2の構造化メタデータとの双方に重複した記述子及び／または記述子群の内、記述内容量の少ない方の前記記述子及び／または記

述子群を消去し、前記第 1 の構造化メタデータに付加された参照記述子と、前記第 2 の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【請求項 13】請求項 5 記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、

前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとの双方に重複した記述子及び／または記述子群の内、前記第 1 の構造化メタデータ側にある記述内容量の少ない方の前記記述子及び／または記述子群を消去し、前記第 1 の構造化メタデータに付加された参照記述子と、前記第 2 の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【請求項 14】請求項 6 記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、

前記第 1 の構造化メタデータに付加された参照記述子により複数指定された参照先の第 2 の構造化メタデータの内の一つを選択して、

前記第 1 の構造化メタデータに付加された参照記述子と、前記第 2 の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第 1 の構造化メタデータとその選択した前記第 2 の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【請求項 15】請求項 1～6 のいずれか一つに記載の構造化メタデータの分割方法において、

前記第 1 の構造化メタデータに前記第 2 の構造化メタデータに関する情報を付加することを特徴とする構造化メタデータの分割方法。

【請求項 16】請求項 1～6 のいずれか一つに記載の構造化メタデータの分割方法において、

前記第 1 の構造化メタデータに前記第 2 の構造化メタデータに関する情報を付加すると共に、前記第 2 の構造化メタデータに前記第 1 の構造化メタデータに関する情報を付加することを特徴とする構造化メタデータの分割方法。

【請求項 17】請求項 15 記載の構造化メタデータの分割方法において、

前記第 1 の構造化メタデータに、前記参照記述子と前記第 2 の構造化メタデータに関する情報とを含む統合先情報記述子群を設けることを特徴とする構造化メタデータの分割方法。

【請求項 18】請求項 16 に記載の構造化メタデータの

分割方法において、

前記第 1 の構造化メタデータに、前記参照記述子と前記第 2 の構造化メタデータに関する情報とを含む統合先情報記述子群を設けると共に、前記第 2 の構造化メタデータに、前記参照記述子と前記第 1 の構造化メタデータに関する情報とを含む統合先情報記述子群を設けたことを特徴とする構造化メタデータの分割方法。

【請求項 19】請求項 1～6、15～18 のいずれか一つに記載の構造化メタデータの分割方法において、

前記第 1 の構造化メタデータの先頭に、この第 1 の構造化メタデータの属性を示す情報を付加すると共に、前記第 2 の構造化メタデータの先頭にこの第 2 の構造化メタデータの属性を示す情報を付加することを特徴とする構造化メタデータの分割方法。

【請求項 20】請求項 19 に記載の構造化メタデータの分割方法において、

前記第 1 の構造化メタデータの属性を示す情報及び前記第 2 の構造化メタデータの属性を示す情報は、伝送時の優先度を示す情報を含むことを特徴とする構造化メタデータの分割方法。

【請求項 21】請求項 20 に記載の構造化メタデータの分割方法により分割した構造化メタデータの伝送方法であって、

各構造化メタデータの繰り返し伝送頻度を、前記伝送時の優先度を示す情報に応じた伝送頻度とすることを特徴とする構造化メタデータの伝送方法。

【請求項 22】請求項 20 に記載の構造化メタデータの分割方法により分割した構造化メタデータの伝送方法であって、

各構造化メタデータを、前記伝送時の優先度を示す情報を用いて伝送時の優先度の高い順に伝送することを特徴とする構造化メタデータの伝送方法。

【請求項 23】請求項 15～18 のいずれか一つに記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、

前記第 1 の構造化メタデータに付加された前記第 2 の構造化メタデータに関する情報を用いて、前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとを統合するか否かを判断し、

統合すると判断された場合には、前記第 1 の構造化メタデータに付加された参照記述子と、前記第 2 の構造化メタデータに付加された参照記述子との内の少なくともどちらか一方の参照記述子を用いて、前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【請求項 24】請求項 15 に記載の構造化メタデータの分割方法において、

前記第 2 の構造化メタデータに関する情報は、前記第 1 の構造化メタデータへの統合の必要性を示す情報を含むことを特徴とする構造化メタデータの分割方法。

【請求項 25】請求項 16 に記載の構造化メタデータの分割方法において、

前記第 2 の構造化メタデータに関する情報は、前記第 1 の構造化メタデータへの統合の必要性を示す情報を含み、前記第 1 の構造化メタデータに関する情報は、前記第 2 の構造化メタデータへの統合の必要性を示す情報を含むことを特徴とする構造化メタデータの分割方法。

【請求項 26】請求項 1～6, 15, 16, 24, 25 のいずれか一つに記載の構造化メタデータの分割方法において、

前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとの双方に前記参照記述子を付加する場合には、少なくともどちらか一方の構造化メタデータにおいてはその構造化メタデータの先頭に参照宣言部を設け前記参照宣言部に前記参照記述子を付加し、

前記第 1 の構造化メタデータのみ前記参照記述子を付加する場合には、前記第 1 の構造化メタデータの先頭に参照宣言部を設け前記参照宣言部に前記参照記述子を付加する、ことを特徴とする構造化メタデータの分割方法。

【請求項 27】請求項 6 に記載の構造化メタデータの分割方法において、

前記第 1 の構造化メタデータの先頭に参照宣言部を設け、前記参照宣言部に前記複数の参照記述子をそれぞれ識別記号を付して付加すると共に、

前記複数の参照記述子が指定する各参照先の前記第 2 の構造化メタデータに関する情報を、その参照先に対応する前記識別記号と共に、前記第 1 の構造化メタデータに付加する、ことを特徴とする構造化メタデータの分割方法。

【請求項 28】請求項 27 に記載の構造化メタデータの分割方法において、

前記第 2 の構造化メタデータに関する情報は、前記第 1 の構造化メタデータへの統合の必要性を示す情報を含むことを特徴とする構造化メタデータの分割方法。

【請求項 29】請求項 27 に記載の構造化メタデータの分割方法において、

前記第 2 の構造化メタデータに関する情報は、前記参照宣言部に付加されることを特徴とする構造化メタデータの分割方法。

【請求項 30】請求項 27 に記載の構造化メタデータの分割方法において、

前記第 2 の構造化メタデータに関する情報は、前記参照宣言部に付加されると共に、前記第 1 の構造化メタデータに設けられる統合先情報記述子群に前記識別記号と共に付加されることを特徴とする構造化メタデータの分割方法。

【請求項 31】請求項 19 または 20 に記載の構造化メタデータの分割方法において、

前記第 1 の構造化メタデータと前記第 2 の構造化メタデ

ータとの双方に前記参照記述子を付加する場合には、少なくともどちらか一方の構造化メタデータにおいて、その構造化メタデータの属性を示す情報の付加位置の後に参照宣言部を設け、前記参照宣言部に前記参照記述子を付加し、

前記第 1 の構造化メタデータのみ前記参照記述子を付加する場合には、前記第 1 の構造化メタデータにおいて、前記第 1 の構造化メタデータの属性を示す情報の付加位置の後に参照宣言部を設け、前記参照宣言部に前記参照記述子を付加する、ことを特徴とする構造化メタデータの分割方法。

【請求項 32】請求項 24, 25, 28 のいずれか一つに記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、

前記第 1 の構造化メタデータに付加された前記第 2 の構造化メタデータに関する情報内の前記第 1 の構造化メタデータへの統合の必要性を示す情報を用いて、前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとを統合するか否かを判断し、

統合すると判断された場合には、前記第 1 の構造化メタデータに付加された参照記述子と、前記第 2 の構造化メタデータに付加された参照記述子との内の少なくともどちらか一方の参照記述子を用いて、前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【請求項 33】請求項 27 または 30 に記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、

前記第 1 の構造化メタデータ内の前記参照宣言部から前記複数の参照記述子と前記識別記号とを得て、

前記第 1 の構造化メタデータから、前記識別記号に基づき前記第 2 の構造化メタデータに関する情報を得て、この前記第 2 の構造化メタデータに関する情報を用いて、前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとを統合するか否かを判断し、

統合すると判断された場合には、前記第 1 の構造化メタデータに付加された参照記述子と、前記第 2 の構造化メタデータに付加された参照記述子との内の少なくともどちらか一方の参照記述子を用いて、前記第 1 の構造化メタデータと前記第 2 の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、デジタルTV放送等におけるコンテンツ内容を記述した構造化メタデータの分割方法、伝送方法、及び統合方法に関するものである。

【0002】

【従来の技術】AVコンテンツの内容を記述するための枠組みとして、MPEG-7 の標準化作業が進められて

いる。MPEG-7とは、AVコンテンツを時間的及び空間的に構造化し、AVコンテンツの内容を木構造を用いて記述する構造化メタデータの規格である。

【0003】MPEG-7記述は、Descriptor (D)とDescriptionScheme (DS)により構成される。Dはコンテンツの特徴を記述するための記述子であり、DSは構造を記述するためのもの(内容記述を記述子間の関係によって構成する方法の規定)である。DSは、Dやより下位のDSを要素として持つことができ、記述子群に相当する。また、これらのDやDSを規定するのが、Description Definition Language (DDL)である。

【0004】MPEG-7記述のフォーマットには、XML (Extensible Markup Language)を用いる。また、DDLはXML Schemaを使って記述する。

【0005】

【発明が解決しようとする課題】AVコンテンツ内容の検索の高速化、効率化のためには、前述した木構造を用いてコンテンツ内容を記述した構造化メタデータが必要である。

【0006】しかし、現段階におけるMPEG-7標準化作業では、1つのMPEG-7記述は1つのファイルで構成されており、そのため、例えばデジタルTV放送において、1チャンネル内の1日分の放送内容のMPEG-7記述(構造化メタデータ)を作るとそのファイルは非常に巨大になり、伝送効率が悪い。

【0007】伝送効率を上げるために単純にMPEG-7記述のファイルを分割したのでは、受信側で分割されたファイルを統合して元の構造化メタデータを再現できない。単なる分割、統合ではMPEG-7記述が断片情報となってしまう、コンテンツの内容記述情報として使用できなくなる。さらに、単なる分割、統合では、上位階層の情報、下位階層の情報といった、コンテンツの内容記述の詳細度に応じた分割、統合ができない。

【0008】また、受信不良や視聴者によるチャンネルの切り替え等によりMPEG-7記述ファイルに欠落情報が発生した場合、欠落情報部分を部分的に再度補完することができず不便であった。

【0009】そこで本発明は、構造化メタデータを再統合可能なように複数のファイルに分割し、その伝送において、分割されたファイルごとに伝送の頻度を制御することにより伝送効率を高めることを可能とする構造化メタデータの分割方法、伝送方法、及び統合方法を提供することを目的としている。さらに本発明は、受信側において欠落データが発生した場合にも、不完全となるファイルの大きさを小さくとどめることができるとともに、不完全なファイルのみを再度補間できる構造化メタデータの分割方法、伝送方法、及び統合方法を提供すること

を目的としている。

【0010】

【課題を解決するための手段】そこで、上記課題を解決するために本発明は、以下の構造化メタデータの分割方法、伝送方法、及び統合方法を提供するものである。

(1) コンテンツの内容を記述するために、記述子と記述子群をノードとして木構造で構成した構造化メタデータを、第1の構造化メタデータとその第1の構造化メタデータに対して下位の関係となる第2の構造化メタデータとに分割する方法であって、第1の構造化メタデータと第2の構造化メタデータとの双方に、または、前記第1の構造化メタデータだけに、この2つの構造化メタデータの接続点を示す情報である参照記述子を付加して分割することを特徴とする構造化メタデータの分割方法。

(2) 上記(1)記載の構造化メタデータの分割方法において、前記第2の構造化メタデータに前記第1の構造化メタデータに含まれる記述子群に対応するダミーの記述子群を付加して分割し、前記第1の構造化メタデータに付加する参照記述子は、前記ダミーの記述子群を接続点とするものであり、前記第2の構造化メタデータに参照記述子を付加する場合のその参照記述子は、前記ダミーの記述子群の要素として付加され、その付加される参照記述子は前記ダミーの記述子群に対応する前記第1の構造化メタデータに含まれる記述子群を接続点とするものである、ことを特徴とする構造化メタデータの分割方法。

(3) 上記(1)または(2)記載の構造化メタデータの分割方法において、前記第1の構造化メタデータと前記第2の構造化メタデータとで、同一の記述子及び同一の記述子群の内の少なくとも一方を重複させて分割することを特徴とする構造化メタデータの分割方法。

(4) 上記(3)記載の構造化メタデータの分割方法において、前記第1の構造化メタデータと前記第2の構造化メタデータとに重複させる記述子及び/または記述子群は、一方の構造化メタデータ側において、他方の構造化メタデータ側よりも記述内容量が少ないことを特徴とする構造化メタデータの分割方法。

(5) 上記(4)記載の構造化メタデータの分割方法において、前記第1の構造化メタデータと前記第2の構造化メタデータとに重複させる記述子及び/または記述子群は、前記第1の構造化メタデータ側において、前記第2の構造化メタデータ側よりも記述内容量が少ないことを特徴とする構造化メタデータの分割方法。

(6) 上記(1)～(5)のいずれか一つに記載の構造化メタデータの分割方法において、前記第1の構造化メタデータに付加する参照記述子は、前記第2の構造化メタデータの存在場所を示す参照先を複数記述したものであることを特徴とする構造化メタデータの分割方法。

(7) 上記(1)～(6)のいずれか一つに記載の構

造化メタデータの分割方法により分割した構造化メタデータの伝送方法であって、前記第1の構造化メタデータの繰り返し伝送頻度を、前記第2の構造化メタデータの繰り返し伝送頻度よりも高くしたことを特徴とする構造化メタデータの伝送方法。

(8) 上記(1)～(6)のいずれか一つに記載の構造化メタデータの分割方法により分割した構造化メタデータの伝送方法であって、前記第1の構造化メタデータの伝送路を、前記第2の構造化メタデータの伝送路よりも伝送品質の高い伝送路としたことを特徴とする構造化メタデータの伝送方法。

(9) 上記(1)記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

(10) 上記(2)記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

(11) 上記(3)記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、前記第1の構造化メタデータと前記第2の構造化メタデータとの双方に重複した記述子及び／または記述子群を、前記第1の構造化メタデータまたは前記第2の構造化メタデータから消去し、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

(12) 上記(4)記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、前記第1の構造化メタデータと前記第2の構造化メタデータとの双方に重複した記述子及び／または記述子群の内、記述内容量の少ない方の前記記述子及び／または記述子群を消去し、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

(13) 上記(5)記載の構造化メタデータの分割方

法により分割された構造化メタデータの統合方法であって、前記第1の構造化メタデータと前記第2の構造化メタデータとの双方に重複した記述子及び／または記述子群の内、前記第1の構造化メタデータ側にある記述内容量の少ない方の前記記述子及び／または記述子群を消去し、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

(14) 上記(6)記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、前記第1の構造化メタデータに付加された参照記述子により複数指定された参照先の第2の構造化メタデータの内の一つを選択して、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくとも一方の参照記述子を用いて、前記第1の構造化メタデータとその選択した前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

(15) 上記(1)～(6)のいずれか一つに記載の構造化メタデータの分割方法において、前記第1の構造化メタデータに前記第2の構造化メタデータに関する情報を付加することを特徴とする構造化メタデータの分割方法。

(16) 上記(1)～(6)のいずれか一つに記載の構造化メタデータの分割方法において、前記第1の構造化メタデータに前記第2の構造化メタデータに関する情報を付加すると共に、前記第2の構造化メタデータに前記第1の構造化メタデータに関する情報を付加することを特徴とする構造化メタデータの分割方法。

(17) 上記(15)記載の構造化メタデータの分割方法において、前記第1の構造化メタデータに、前記参照記述子と前記第2の構造化メタデータに関する情報とを含む統合先情報記述子群を設けることを特徴とする構造化メタデータの分割方法。

(18) 上記(16)記載の構造化メタデータの分割方法において、前記第1の構造化メタデータに、前記参照記述子と前記第2の構造化メタデータに関する情報とを含む統合先情報記述子群を設けると共に、前記第2の構造化メタデータに、前記参照記述子と前記第1の構造化メタデータに関する情報とを含む統合先情報記述子群を設けたことを特徴とする構造化メタデータの分割方法。

(19) 上記(1)～(6)、(15)～(18)のいずれか一つに記載の構造化メタデータの分割方法において、前記第1の構造化メタデータの先頭に、この第1の構造化メタデータの属性を示す情報を付加すると共に、前記第2の構造化メタデータの先頭にこの第2の構

10

20

30

40

50

造化メタデータの属性を示す情報を付加することを特徴とする構造化メタデータの分割方法。

(20) 上記(19)に記載の構造化メタデータの分割方法において、前記第1の構造化メタデータの属性を示す情報及び前記第2の構造化メタデータの属性を示す情報は、伝送時の優先度を示す情報を含むことを特徴とする構造化メタデータの分割方法。

(21) 上記(20)に記載の構造化メタデータの分割方法により分割した構造化メタデータの伝送方法であって、各構造化メタデータの繰り返し伝送頻度を、前記伝送時の優先度を示す情報に応じた伝送頻度とすることを特徴とする構造化メタデータの伝送方法。

(22) 上記(20)に記載の構造化メタデータの分割方法により分割した構造化メタデータの伝送方法であって、各構造化メタデータを、前記伝送時の優先度を示す情報を用いて伝送時の優先度の高い順に伝送することを特徴とする構造化メタデータの伝送方法。

(23) 上記(15)～(18)のいずれか一つに記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、前記第1の構造化メタデータに付加された前記第2の構造化メタデータに関する情報を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合するか否かを判断し、統合すると判断された場合には、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくともどちらか一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

(24) 上記(15)に記載の構造化メタデータの分割方法において、前記第2の構造化メタデータに関する情報は、前記第1の構造化メタデータへの統合の必要性を示す情報を含むことを特徴とする構造化メタデータの分割方法。

(25) 上記(16)に記載の構造化メタデータの分割方法において、前記第2の構造化メタデータに関する情報は、前記第1の構造化メタデータへの統合の必要性を示す情報を含み、前記第1の構造化メタデータに関する情報は、前記第2の構造化メタデータへの統合の必要性を示す情報を含むことを特徴とする構造化メタデータの分割方法。

(26) 上記(1)～(6)、(15)、(16)、

(24)、(25)のいずれか一つに記載の構造化メタデータの分割方法において、前記第1の構造化メタデータと前記第2の構造化メタデータとの双方に前記参照記述子を付加する場合には、少なくともどちらか一方の構造化メタデータにおいてはその構造化メタデータの先頭に参照宣言部を設け前記参照宣言部に前記参照記述子を付加し、前記第1の構造化メタデータのみ前記参照記述子を付加する場合には、前記第1の構造化メタデータ

の先頭に参照宣言部を設け前記参照宣言部に前記参照記述子を付加する、ことを特徴とする構造化メタデータの分割方法。

(27) 上記(6)に記載の構造化メタデータの分割方法において、前記第1の構造化メタデータの先頭に参照宣言部を設け、前記参照宣言部に前記複数の参照記述子をそれぞれ識別記号を付して付加すると共に、前記複数の参照記述子が指定する各参照先の前記第2の構造化メタデータに関する情報を、その参照先に対応する前記識別記号と共に、前記第1の構造化メタデータに付加する、ことを特徴とする構造化メタデータの分割方法。

(28) 上記(27)に記載の構造化メタデータの分割方法において、前記第2の構造化メタデータに関する情報は、前記第1の構造化メタデータへの統合の必要性を示す情報を含むことを特徴とする構造化メタデータの分割方法。

(29) 上記(27)に記載の構造化メタデータの分割方法において、前記第2の構造化メタデータに関する情報は、前記参照宣言部に付加されることを特徴とする構造化メタデータの分割方法。

(30) 上記(27)に記載の構造化メタデータの分割方法において、前記第2の構造化メタデータに関する情報は、前記参照宣言部に付加されると共に、前記第1の構造化メタデータに設けられる統合先情報記述子群に前記識別記号と共に付加されることを特徴とする構造化メタデータの分割方法。

(31) 上記(19)または(20)に記載の構造化メタデータの分割方法において、前記第1の構造化メタデータと前記第2の構造化メタデータとの双方に前記参照記述子を付加する場合には、少なくともどちらか一方の構造化メタデータにおいて、その構造化メタデータの属性を示す情報の付加位置の後に参照宣言部を設け、前記参照宣言部に前記参照記述子を付加し、前記第1の構造化メタデータのみ前記参照記述子を付加する場合には、前記第1の構造化メタデータにおいて、前記第1の構造化メタデータの属性を示す情報の付加位置の後に参照宣言部を設け、前記参照宣言部に前記参照記述子を付加する、ことを特徴とする構造化メタデータの分割方法。

(32) 上記(24)、(25)、(28)のいずれか一つに記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、前記第1の構造化メタデータに付加された前記第2の構造化メタデータに関する情報内の前記第1の構造化メタデータへの統合の必要性を示す情報を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合するか否かを判断し、統合すると判断された場合には、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくともどちらか一方の参照記述子を用いて、前

記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

(33) 上記(27)または(30)に記載の構造化メタデータの分割方法により分割された構造化メタデータの統合方法であって、前記第1の構造化メタデータ内の前記参照宣言部から前記複数の参照記述子と前記識別記号とを得て、前記第1の構造化メタデータから、前記識別記号に基づき前記第2の構造化メタデータに関する情報を得て、この前記第2の構造化メタデータに関する情報を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合するか否かを判断し、統合すると判断された場合には、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくともどちらか一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合することを特徴とする構造化メタデータの統合方法。

【0011】

【発明の実施の形態】《MPEG-7記述ファイルの分割と統合》MPEG-7記述ファイルの分割方法として、まず、以下の2種類を説明する。

- (1) MPEG-7記述ファイル中の任意のDSをルートとして、そのDS以下を元のMPEG-7記述から分離する。
- (2) MPEG-7記述ファイル中の任意のDS以下にある1つまたは複数のD/DS（とそれ以下の要素）をまとめて分離し、ルートとなるダミーのDSの下に付ける。

【0012】それぞれの分割方式の例を図1、図2に示す。図1ではDS3を新たなルートとして、それ以下の要素を分離している。また、図2ではDS2の下に2つの要素D2、DS3とDS3以下の要素を分離し、ダミーのDSを新たなルートとしてその下に分離した要素を付けている。

【0013】分割した複数のMPEG-7記述を再統合して1つのMPEG-7記述を得るためには、分割する際に接続点についての情報を記述の中に埋め込む必要がある。そこで、分離した側（下位ファイルである第2の構造化メタデータ側）の新たなルートの下に分離前に付いていたDSへの参照（参照記述子）を追加し、また分離元の側（上位ファイルである第1の構造化メタデータ側）のDSの下にも分離した新たなルートへの参照（参照記述子）を付ける。例えば、図1では分離した側のDS3の下に分離元であるDS2への参照を付けるとともに、分離元の側のDS2の下にはDS3への参照を付ける。また、図2では分離の際に作成された分離した側のダミーのDSの下に分離元であるDS2への参照を付けるとともに、分離元の側のDS2の下にはダミーのDSへの参照を付ける。このように接続点の両方の側に参照を付けること（即ち、第1の構造化メタデータと第2の構造化メタデータとの双方に、この2つの構造化メタデータの接続点を示す情報である参照記述子を付加すること）により、柔軟な統合処理を行うこ

とができるようになる。なお、接続点の片側である分離元の側のみに参照を付ける（即ち、第1の構造化メタデータのみに、この2つの構造化メタデータの接続点を示す情報である参照記述子を付加する）ようにしてもよい。

【0014】分離された2つのファイルを統合する場合には、

(イ) 分離元の側（上位ファイルである第1の構造化メタデータ側）と、分離した側（下位ファイルである第2の構造化メタデータ側）との双方に参照記述子が付されている場合には、少なくとも一方の側の参照記述子を用いて、2つのファイルを統合する。

(ロ) 分離元の側（上位ファイルである第1の構造化メタデータ側）のみに参照記述子が付されている場合には、その参照記述子を用いて、2つのファイルを統合する。

《再接続のためのDS》

【Connection DS】接続点における接続先への参照を記述するためのDSとして、統合先情報記述子群であるConnection DSを定義する。Connection DSのDDL定義は図11に示すようになる。

【0015】Connection DSはその要素にReference D（参照記述子）を持ち、参照先をURLで指定することができる。さらに、Reference Dを複数持つことができるので、参照先を複数定義することができる。これは例えばデジタル放送において、1つのReference Dはデジタル放送ストリーム内の接続先ノードを参照し、もう1つのReference Dはインターネット上に存在する接続先を参照するといったことができる。この例を図10に示す。これにより、仮にデジタル放送ストリーム内の接続先のデータを取得できなかったとしても、それに代わるデータをインターネット上から取得できるようになる。

【0016】このように、上位ファイルに付加されたConnection DSにおけるReference D（参照記述子）により、下位ファイルの参照先が複数指定されている場合には、複数指定された参照先の下位ファイルの内の一つを選択して、上位ファイルとその選択した下位ファイルとを統合するまた、Connection DSはその要素に任意のD/DSを任意の数だけ持つことができる。これについては後で説明する。

【0017】属性“type”は“up”または“down”のいずれかの値を取り、参照先との接続の方法を指定する。

属性“type”が“up”である場合は自身が分離してきた新しいルートであり、参照先が分離前に所属していたDSであることを示す。また、属性“type”が“down”である場合は参照先が分離してできた新しいルートであることを示す。

【0018】図1において接続先への参照にConnection DSを用いた例を図3に示す。DS3の下にConnection DS

の属性 "type" は "up" となり、DS2の下でConnection DSの属性 "type" は "down" となる。

【0019】ファイルを分割する際に、分割ノードの付近の一部のD及び／またはDSを両方のファイルに重複して含ませたいという要求が考えられる。例えば、分割したファイルの親側か子供側のどちらの一方を入手した場合でも利用したいような重要な情報があれば、それは両方のファイルに重複して含ませるのが望ましい。これを実現するために、Connection DSの要素に重複させたいD/DSを任意の数だけ持たせることができるように定義した。これを用いて情報を重複させてファイルを分割する例を図4に示す。この例では、DS3とD3が両方のファイルに重複して存在している。

【0020】また、重複させるD及び／またはDSの記述内容について、一方の分割ファイルに含ませるものは簡略化した内容（記述内容量の少ない内容）とし、他方の分割ファイルに含ませるものはより詳細な内容とすることで、重複に伴う情報量の冗長性を抑えることができる。これについて、図13に示す。この例では、DS3とD3について内容を簡略化したDS3'とD3'を上位ファイルに含ませ、詳細な内容のままのDS3とD3を下位ファイルに含ませている。

【0021】図4に示す分割ファイルを統合する場合には、上位ファイルに付加したConnection DSの下に付いているDS3とD3とを、両方のファイルに重複して存在しているものと判断し、上位ファイル側のConnection DSの下に付いているDS3とD3とを消去して、上位ファイルと下位ファイルとをConnection DSに応じて統合する。なお、両方のファイルに重複しているD/DSがある場合の2つのファイルの統合時、上位ファイル側に存在している重複D/DSを消去するか、下位ファイル側に存在している重複D/DSを消去するかは、状況に応じて優先度を決めて実行してもよい。

【0022】図13に示す分割ファイルを統合する場合には、上位ファイルの簡略化したDS3'とD3'とを消去してConnection DSに応じて統合する。この際、重複しているD/DS同士の情報量を比較、即ちDS3とDS3'、及び、D3'とD3の情報量をそれぞれ比較して、簡略化した内容であるDS3'とD3'を消去対象として選択する。また、ファイル分割時に簡略化した内容のD/DSを含ませる場合は、必ず上位ファイルに含ませるとするルールを設定することで、上位ファイルと下位ファイルとで重複したD/DSの情報量の比較無しに、DS3'とD3'を消去対象として選択することができる。

【0023】Connection DSを実際に使用するためには、MPEG-7の全てのDSが要素としてConnection DSを持てなければならない。そこで、MPEG-7の全てのDSのオプションの要素としてConnection DSを加えるようにする。

【Temporary DS】分割方法(2)で用いるダミーのDSとし

て、Temporary DSというものを新たに導入する。Temporary DSのDDL定義は図12に示すようになる。

【0024】Temporary DSは要素にConnection DSを持つとともに、任意のD/DSを持つ。

【0025】なお、ここまで説明したMPEG-7記述ファイルの分割方法、統合方法は適宜組み合わせ用いてもよい。分割方法の組み合わせとしては、例えば、図2に示す方法と図3（図10）に示す方法との組み合わせ、図2に示す方法と図4に示す方法との組み合わせ、図2に示す方法と図3（図10）に示す方法と図4に示す方法との組み合わせ、図3（図10）に示す方法と図4に示す方法との組み合わせが考えられる。統合方法の組み合わせは、この分割方法の組み合わせに合わせて適宜組み合わせ用いればよい。また、ファイルの分割時に付加したConnection DS、Temporary DS等の接続先を示す情報は、ファイルの統合時に削除するようにしてもよい。

《分割ファイル伝送のためのシステム》デジタル放送などの伝送系のシステムにおいて分割されたMPEG-7記述ファイルを伝送する場合、複数のファイルを1つのMPEG-7ストリームにマルチプレクス：Multiplexして伝送するのが望ましい。また、分割されたファイルごとにコンテンツとの同期の取り方やカルーセル伝送の頻度などを変えた方が、より柔軟なMPEG-7ストリームを構成することができる。

【0026】そこで、図5に示すMPEG-7伝送システムを新たに導入する。MPEG-7記述は、上述または後述の分割方法に従って "Divider" によって複数のファイルに分割される。分割された各ファイルは "MPEG-7 encoder" によって符号化され、 "Packetizer" によってパケット化される。 "Transport controller" は、各ファイルの同期の取り方やカルーセル伝送の頻度などの基準に従ってパケットの出力を制御する。最後に "Multiplexer" がパケットをマルチプレクス：Multiplexし、MPEG-7ストリームを生成する。

【0027】このMPEG-7ストリームを受信するためのシステムは図6のようになる。MPEG-7ストリームはまず "Demultiplexer" によってデマルチプレクス：Demultiplexされる。次に、ファイルごとに "MPEG-7 decoder" によってデコードされる。最後に各ファイルを、上述または後述の統合方法に従って "Synthesizer" によって合成し、元のMPEG-7記述を得る。

【0028】なお、受信システムの "Demultiplexer" によってMPEG-7ストリームをデマルチプレクス：Demultiplexする際、各分割ファイルに対応するパケットを識別できなければならない。そこで、一つの方法として、各パケットのヘッダーにファイルを一意に識別するIDを付けるようにする。

《Broadcasting デジタル放送への応用例》デジタル放送でコンテンツと同時にMPEG-7データ記述を送信す

10

20

30

40

50

ることを想定する。また、視聴者が見ているか、録画しているチャンネルに対応するMPEG-7データしか取得できないと仮定する。すなわち、視聴者がチャンネルを切り替えると、その時点でMPEG-7データが途切れてしまうと仮定する。

【0029】ここで、MPEG-7記述として送信する内容について、以下のように分類する。

(1)チャンネル内の1日の番組構成 (番組表)

(2)番組内のコーナー構成

(3)コーナー内の詳細情報

このとき、(1)の情報はもっとも重要な情報であり、1日を通してカルーセル伝送する必要がある。(2)の情報は、その次に重要な情報であり、番組を通してカルーセル伝送する必要がある。(3)の情報はもっとも重要度の低い情報であり、コーナー内のみで伝送されればよい。そこで、MPEG-7記述を上分類に基づいて図7のように分割する。そして、分割ファイルごとにカルーセル伝送の頻度を制御する ((1)の情報である上位のファイルの伝送頻度を、(2)の情報である下位のファイルの伝送頻度よりも高くする、同様に(2)の情報の伝送頻度を(3)の情報の伝送頻度よりも高くする) ことにより、上の要求を実現することができる。

【0030】さらに、視聴者がチャンネルを切り替えたなどの原因により、コーナー内の詳細情報を取得できなかったと仮定する。このような場合でも、コーナー内の詳細情報と同一内容のデータをインターネット上に用意しておき、番組内のコーナー情報からコーナー内の詳細情報を参照するConnection DSIに、デジタル放送ストリーム上のコーナー内詳細情報への参照とともにインターネット上の同一情報への参照も記述しておけば、欠落データをインターネットからダウンロードして補うことができる。

【0031】この例を図8に示す。放送局は、コンテンツに関するMPEG-7記述を作成し、上述した方法で複数のファイルに分割する。分割されたファイルはデジタル放送でMPEG-7ストリームとして放送されるとともに、インターネットに接続されたMPEG-7データベースにも保存される。このとき、上位ファイルに付加するConnection DSIにおいて、下位ファイルへの参照先を2つ用意し、1つはデジタル放送で放送されるMPEG-7ストリーム上の接続先を参照先とし、他の1つはMPEG-7データベース上の接続先を参照先とする (図10参照)。

【0032】視聴者側では、デジタル放送を通じてMPEG-7ストリームを受信し、MPEG-7デコーダー (図6に示すDemultiplexer, MPEG-7 decoder, Synthesizerをまとめて図8ではMPEG-7 decoderとした) でデコードし、分割されたファイルを1つに統合する。このとき、視聴者によるチャンネルの切り替え等の原因で欠落したファイルがあれば、その上位ファイルに書かれた

参照情報のうち、インターネット上のMPEG-7データベースにある接続先への参照に基づいて、MPEG-7データベースより欠落ファイルに相当するファイルを取得し、それを代わりに用いる。統合されたMPEG-7記述は様々なMPEG-7アプリケーションで使うことができる。

【0033】なお、上位ファイルの伝送路を、下位ファイルの伝送路よりも伝送品質の高い伝送路とするようにしてもよい。

10 【0034】また、他の応用例を図9に示す。図9において、放送局は、コンテンツに関するMPEG-7記述を作成し、デジタル放送でコンテンツとともに放送し、また、インターネットに接続されたビデオ・データベースにコンテンツとMPEG-7記述とを保存する。

【0035】視聴者側では、デジタル放送を受信し、"Demultiplexer"がコンテンツ (ビデオ・ストリーム) とMPEG-7ストリームとを分離する。ビデオ・ストリームは視聴者側のデータベースに保存される。MPEG-7ストリームはMPEG-7デコーダーによつてデコードされ、MPEG-7記述が得られる。

20 【0036】ここで、視聴者によるチャンネル切り替え等の何らかの理由でビデオ・ストリームの一部が欠落していた場合、ビデオ補完器 (Supplementer) はMPEG-7記述を利用してインターネット上のビデオ・データベースを検索し、同一内容のビデオ・ストリームを見つける。そして、MPEG-7記述のセグメント情報を利用して欠落部分を含むセグメントのみをビデオ・データベースから取得し、それを視聴者側のビデオ・データベースにあるビデオ・ストリームに接続することにより、欠落部分を補完する。

【0037】次に、接続点を示す情報 (参照記述子) に加えて、各分割したファイル (各分割した構造化メタデータ) 自体の属性を示す情報や統合先のファイルの情報を付加して分割する方法、及びそれらの情報を利用して統合する方法について説明する。

《Mpeg7Main、Mpeg7Unitの説明》MPEG-7では、記述ファイルの先頭の要素であるルート・エレメントとして、Mpeg7MainとMpeg7Unitの2つを定義している。

40 【0038】ルート・エレメントがMpeg7MainであるファイルはMPEG-7記述として妥当である。MPEG-7パーサーはそのファイルを読み、MPEG-7記述として解釈することができる。Mpeg7Mainの子の要素として記述できるものは、一部の要素に限定される。

【0039】一方、ルート・エレメントがMpeg7UnitであるファイルはMPEG-7記述の単なる断片である。MPEG-7パーサーはそのファイルのみではMPEG-7記述として解釈することができない。他のファイルのConnection DSIから参照される

ことにより、MPEG-7記述として解釈することができる。

【0040】図1に示した分割方法(1)による分割では、分割前のファイルのルート・エレメントがMpeg7Mainであれば分割後の上位ファイルのルート・エレメントもMpeg7Mainとなり、分割前のファイルのルート・エレメントがMpeg7Unitであれば分割後の上位ファイルのルート・エレメントもMpeg7Unitとなる。分割後の下位ファイルについては、下位ファイルの先頭の要素によって異なる。先頭の要素がMpeg7Mainの子要素として記述できるものであれば、下位ファイルのルート・エレメントはMpeg7Mainとなる。先頭の要素がMpeg7Mainの子要素として記述できないものであれば、下位ファイルのルート・エレメントはMpeg7Unitとなる。

【0041】図2に示した分割方法(2)による分割では、分割後の上位ファイルのルート・エレメントは、分割方法(1)の場合と同様に分割前のファイルのルート・エレメントと同一である。分割後の下位ファイルについては、ルート・エレメントは必ずMpeg7Unitとなる。

《属性の説明》MPEG-7記述ファイル(構造化メタデータ)の分割の際に、統合先情報記述子群であるConnectionDSの属性として、統合先ファイル(接続先ファイル)に関する情報を付加する。よって、分割後の上位ファイルには統合先(接続先)である下位ファイルに関する情報が付加され、分割後の下位ファイルには統合先(接続先)である上位ファイルに関する情報が付加されている。なお、上位ファイル側にのみ統合先ファイル(接続先ファイル)に関する情報を付加する
30 ようにしてもよい。

【0042】図14はConnectionDSに付加する属性(接続先ファイルに関する情報)の例である。また、属性を含めたConnectionDSのDDL定義は図15のようになる。属性rootTypeは接続先のファイルがMpeg7Mainであるか、Mpeg7Unitであるかを識別する。属性contentは接続先ファイルの内容の概略を示す。これらの属性情報を利用し、例えば分割された上位ファイルのみを入手したときに、接続先の下位ファイルも入手するか
40 どうかを属性の値によって判断するなどの処理を行うことができる。

【0043】また、分割した各ファイルにおいて、ファイルの先頭の要素であるルート・エレメント(Mpeg7Main及びMpeg7Unit)の属性として、そのファイル自身の属性を示す情報を付加する。図16はルート・エレメントに付加する属性の例である。図17～図19は属性を含めたルート・エレメントのDDL定義である。

【0044】図において、属性lengthはファイル
50

の大きさをバイト単位で記述する。属性priorityはファイルの優先度である。属性contentはファイルの内容の概略を示す。これらの属性情報を利用し、例えばMPEG-7記述の伝送において、優先度の高いファイルから順に伝送するなどの処理を行うことができる。あるいは、ファイルの大きさや内容の概略によってファイルの処理方法を判断することなどができる。

【0045】なお、図17～図19のDDL定義において、Mpeg7RootはMpeg7Main及びMpeg7Unitに共通の抽象型である。Mpeg7Main及びMpeg7Unitに共通の属性は、Mpeg7Rootの属性として定義すればよい。

《インターネットを介したコンテンツ検索における属性の活用例》図20は、インターネットを介したビデオ・コンテンツ検索のネットワーク図である。ビデオ・データベースには、多数のビデオ・コンテンツとそれぞれに対応するMPEG-7記述ファイルが記憶されている。また、MPEG-7記述ファイルはビデオ・セグメント情報と、色や音声などの分野ごとの記述子とに上述の分割方法を用いて分割されている。ビデオ・セグメント情報ファイル内には複数のConnectionDSが存在し、それぞれのConnectionDSが分野ごとの各記述子ファイルを接続先ファイルとするとともに、接続先ファイルの内容の概略が属性として付加されている。ユーザーはインターネットを介してMPEG-7データをダウンロードし、ビデオ・コンテンツの検索を行うことができる。

【0046】図21は、図20に示したインターネットを介したビデオ・コンテンツ検索のフローチャートである。

【0047】ユーザーは、初めに、ビデオ・コンテンツ・データベースからビデオ・セグメント情報ファイルをダウンロードする。次に、ビデオ・セグメント情報内のConnectionDSには接続先ファイルの内容の概略を示す属性が付加されているので、その情報を元にして検索に使用する記述子を含む記述子ファイルをダウンロードする。例えば、色に関する記述子で検索するなら色に関する記述子ファイルをダウンロードし、音声に関する記述子で検索するなら音声に関する記述子ファイルをダウンロードする。また、検索に使用する記述子ファイルへの参照記述子とそのビデオ・セグメント情報に存在しなければ、そのコンテンツはスキップして次のコンテンツの検索に移る。

【0048】ダウンロードされた記述子ファイルは、ユーザーが設定した検索条件とを照らし合わされる。そして、もし検索条件に適合していれば、対応するビデオ・コンテンツをビデオ・コンテンツ・データベースからダウンロードし、検索を終了する。もし検索条件に適合しなければ、次のコンテンツの検索に移る。

《MPEG-7ストリーム構成時の属性の活用例》図2

2は、MPEG-7記述伝送時の優先度を利用したストリーム構成方法を示す図である。

【0049】図22(a)に示すように、MPEG-7記述が6個のファイルに分割されると仮定する。また、分割された各ファイルのルート・エレメントには、伝送の優先度を示す属性priorityが付加されているとする。

【0050】図22(a)に示す複数のMPEG-7記述ファイルを、その構造に従って並べると、図22

(b)のような順番で並べることができる。これを属性priorityの値の大きい順に並べ替えると、図22(c)のようになる。

【0051】図22(c)の順番で伝送することで、受信側は優先度の高いファイルを先に入手することができる。また、何らかの理由で伝送が途中で途切れた場合でも、受信側には優先度の高いファイルが残る。また、受信側の装置の制限等で全てのファイルを受信できない場合でも、優先度の高いファイルから順に受信し、所定の優先度よりも優先度の低いファイルを受信しないといったことが可能となる。

《デジタル放送でのカルーセル伝送における属性の活用例》デジタル放送において、MPEG-7記述を複数のファイルに分割し、ファイルごとに異なる頻度でカルーセル伝送するとき、各ファイルのルート・エレメントに伝送の優先度を示す属性priorityを付加しておけば、それに基づいてカルーセル伝送の頻度を自動的に決定することが可能である。

【0052】ユーザーが各ファイルのカルーセル伝送の頻度を指定する場合は、全体のデータ伝送量を考慮して頻度の調整を行う必要があるが、優先度のみを記述する場合にはそのような調整を行う必要がないので、ユーザーの手間を大きく省くことができる。

《Connection DSの属性の追加》先に、図14で説明したConnection DSに付加する属性(接続先ファイルに関する情報)の追加情報(属性use)について説明する。

【0053】図23は属性useを追加したConnection DSの属性の例である。また、図24は属性useを追加したConnection DSのDDL定義である。属性useは接続先ファイルの必要性を示し、requiredまたはoptionalの値を取る。属性useがrequiredであれば、接続先ファイルが必要不可欠であることを示し、MPEG-7アプリケーションは接続先ファイルを必ず入手しなければならない。属性useがoptionalであれば、接続先ファイルが必ずしも入手しなくてもよいことを示す。この属性useを付加する場合には、少なくとも上位ファイル側に付加するようにする。

【0054】属性useにより、例えば接続先ファイルがある種のヘッダー情報や著作権情報などの必要不可欠

な情報の場合に、属性useをrequiredとすることで、接続先ファイルを入手することをアプリケーションに強制することができる。

【0055】これについてまとめると、前記第1の構造化メタデータ(上位側の構造化メタデータ)へ付加する前記第2の構造化メタデータ(下位側の構造化メタデータ)に関する情報は、前記第1の構造化メタデータへの統合の必要性を示す情報(属性use)を含むようにする。また、前記第2の構造化メタデータへ前記第1の構造化メタデータに関する情報を付加する場合には、前記第1の構造化メタデータに関する情報が前記第2の構造化メタデータへの統合の必要性を示す情報を含むようにしてもよい。

【0056】統合方法については、前記第1の構造化メタデータに付加された前記第2の構造化メタデータに関する情報内の前記第1の構造化メタデータへの統合の必要性を示す情報を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合するかどうかを判断し、統合すると判断された場合には、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくともどちらか一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合する。

《参照宣言の説明》分割されたMPEG-7記述ファイルを入手したときに、そのファイル内のConnection DSが参照しているファイル(接続先ファイル)を知るためには、Connection DSが記述されている所まで構文解析を進めなければならない。多くの分割されたMPEG-7記述ファイルの中から、あるMPEG-7記述ファイルの接続先ファイルとして必要なファイルを選別するときに、接続先ファイルとなる全てのファイルの構文解析を行うのでは効率が悪い。

【0057】そこで、MPEG-7記述ファイル内のConnection DSの参照情報(参照記述子)を、ファイルの先頭に設けた参照宣言部に記述する。これにより、ファイルの先頭部分のみを構文解析するだけで、そのファイル内のConnection DSが参照しているファイル(接続先(統合先)のファイル)を知ることができる。

【0058】なお、MPEG-7記述ファイル自体の属性を示す情報(図16に示す属性)をそのファイルの先頭に付加する場合には、参照宣言部はファイル自体の属性を示す情報の後ろ側に設けるようにする。

【0059】参照宣言部内の各参照情報に参照情報を識別するIDを付加すれば、各Connection DSに参照情報そのものではなくこのIDを記述することにより、接続先ファイルの情報を得ることができる。

【0060】図29は参照宣言部を利用した接続先の参照方法を説明する図である。ファイルfile1.xm

1内のReference Declaration DSが参照宣言部であり、この中に参照情報とそれを識別するIDが列挙されている。ファイルfile1.xml内の各Connection DSには、参照情報を記述する代わりに参照宣言内の参照情報を識別するIDが記述され、これによって接続先を参照する。

【0061】図25、図26、図27は、参照宣言部を導入する場合のConnection DSのDDL定義である。図25はConnection DSの要素として使用するReference DのDDL定義である。Reference Dの属性typeは参照先がそのファイル内(internal)であるか、ファイル外(external)であるかを示す。属性typeがinternalで、参照先の指定に属性idrefを使用する場合、そのIDはファイル内の要素に付加されたIDを指し示す。属性typeがexternalで、参照先の指定に属性idrefを使用する場合、そのIDは参照宣言内の参照情報に付加されたIDを指し示す。

【0062】図26はConnection DS及びReference Declaration DSで共通に用いる属性のDDL定義である。

【0063】図27は、図25のReference Dの定義、及び図26の属性の定義を利用したConnection DSのDDL定義である。

【0064】図28は、Reference Declaration DSのDDL定義である。Reference Declaration DSの要素Itemによって参照情報を列挙する。要素Itemは、属性として、参照先を示すhref及び参照情報を識別するidの他に、Connection DSと同様の属性(rootType、content、use)を持つ。

【0065】前記した参照宣言部を設けたMPEG-7記述ファイルに、図23で説明したような接続先ファイルに関する情報を付加する場合には、少なくとも参照宣言部に付加するようにする。接続先ファイルに関する情報は、さらに、参照情報(参照記述子)を識別するIDと共にConnection DSに付加するようにしてもよい。参照宣言部とConnection DSとの両方に接続先ファイルに関する情報を付加する場合には、同じ内容の情報を付加してもよいが、Connection DS側に付加する接続先ファイルに関する情報の方をより詳しい内容のものとしてもよい。参照宣言部とConnection DSとの両方に接続先ファイルに関する情報が付加されている場合、参照先決定に際しては、Connection DS側に付加されている接続先ファイルに関する情報を優先させる。

【0066】統合方法については、前記第1の構造化メタデータ(上位側の構造化メタデータ)内の前記参照宣

言部から前記複数の参照記述子と前記識別記号とを得て、前記第1の構造化メタデータから、前記識別記号に基づき前記第2の構造化メタデータ(下位側の構造化メタデータ)に関する情報を得て、この前記第2の構造化メタデータに関する情報を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合するかどうかを判断し、統合すると判断された場合には、前記第1の構造化メタデータに付加された参照記述子と、前記第2の構造化メタデータに付加された参照記述子との内の少なくともどちらか一方の参照記述子を用いて、前記第1の構造化メタデータと前記第2の構造化メタデータとを統合するようにする。

【0067】

【発明の効果】以上の通り、本発明の構造化メタデータの分割方法、伝送方法、及び統合方法を用いれば、構造化メタデータを再統合可能なように複数のファイルに分割でき、その伝送において、分割されたファイルごとに伝送を制御することにより伝送効率を高めることができる。さらに本発明を用いれば、受信側において欠落データが発生した場合にも、構造化メタデータが再統合可能なように複数のファイルに分割されているので、不完全となるファイルの大きさを小さくとめることができる。とともに、不完全なファイルのみを再度補間でき、情報の欠落に対する耐性を高めることが可能となる。

【0068】また、本発明を用いれば、上位階層の情報、下位階層の情報といった、コンテンツの内容記述の詳細度に応じた分割、統合が行える。

【0069】ダミーの記述子群を用いるようにした場合には、より多様な分割が可能となり、分割の自由度が広がる。

【0070】第1の構造化メタデータと第2の構造化メタデータとで、同一の記述子及び同一の記述子群の内の少なくとも一方を重複させて分割するようにした場合には、どちらか一方の構造化メタデータのみしか取得できなくとも、重複させた情報を得ることができ、より情報の欠落に対する耐性を高めることが可能となる。

【0071】重複させる記述子及び/または記述子群は、一方の構造化メタデータ側において、他方の構造化メタデータ側よりも記述内容量が少ないものとした場合には、重複に伴う情報量の冗長性を抑えることができる。また、分割した構造化メタデータの統合時において、重複している記述子及び/または記述子群の記述内容の情報量に応じて消去対象を選択することで、より詳細な内容の記述子及び/または記述子群を残すことができる。

【0072】重複させる記述子及び/または記述子群が、第1の構造化メタデータ側において、第2の構造化メタデータ側よりも記述内容量が少ないものとして分割する場合には、分割した構造化メタデータの統合時において、重複している記述子及び/または記述子群同士

記述内容の情報量を比較することなく、重複している記述子及び／または記述子群から消去対象となる記述内容量が少ないものを選択することができる。

【0073】第1の構造化メタデータに付加する参照記述子が、第2の構造化メタデータの存在場所を示す参照先を複数記述したものである場合には、受信側において欠落データが発生しても、複数の経路で不完全なファイルを再度補間することができ、より情報の欠落に対する耐性を高めることが可能となる。

【0074】分割した構造化メタデータに、統合先（接続先）の構造化メタデータに関する情報を付加した場合には、統合先の構造化メタデータを入手することなく統合先メタデータの内容の概略を知ることができ、統合先の構造化メタデータを入手する必要があるかないかを、統合先の構造化メタデータを入手することなく判断できる。

【0075】分割した各構造化メタデータの先頭に、その構造化メタデータ自身の属性を示す情報を付加した場合には、構造化メタデータの先頭を見るだけで、データの大きさや内容の概略等の属性を知ることができる。また、上記構造化メタデータ自身の属性を示す情報が、伝送時の優先度を示す情報を含む場合には、分割した各構造化メタデータの伝送の順番をその優先度に基づいて決定したり、カルーセル伝送の頻度を上記優先度に基づいて決定することができる。

【0076】分割した構造化メタデータに付加した統合先（接続先）の構造化メタデータに関する情報が統合の必要性を示す情報を含む場合には、統合先（接続先）の構造化メタデータを入手することを強制できる。統合先（接続先）の構造化メタデータを著作権保護データとしておけば、統合元（接続元）の構造化メタデータの著作権保護を強化できる。

【0077】分割した構造化メタデータの先頭部分に参照宣言部を設けた場合には、構造化メタデータの先頭部分のみを構文解析するだけで、その構造化メタデータの統合先（接続先）の構造化メタデータを知ることができる。また、参照記述子を一箇所で効率的に管理できるという効果がある。さらに、構造化メタデータ内に同一の統合先（接続先）構造化メタデータに関する情報が複数ある場合でも、各構造化メタデータに関する情報には参照記述子の識別記号を付し、実際の参照記述子は参照宣言部に1つだけ記述すればよいので、構造化メタデータのサイズを小さくできる。

【図面の簡単な説明】

【図1】ファイル分割方法の一実施例を示す図である。

【図2】ファイル分割方法の一実施例を示す図である。

【図3】ファイル分割方法の一実施例を示す図である。

【図4】ファイル分割方法の一実施例を示す図である。

【図5】MPEG-7伝送システムを示す図である。

【図6】MPEG-7受信システムを示す図である。

【図7】ファイル分割方法の一実施例を示す図である。

【図8】Broadcasting デジタル放送への応用例を示す図である。

【図9】Broadcasting デジタル放送への応用例を示す図である。

【図10】ファイル分割方法の一実施例を示す図である。

【図11】Connection DSのDDL定義を示す図である。

【図12】Temporary DSのDDL定義を示す図である。

【図13】ファイル分割方法の一実施例を示す図である。

【図14】一実施例のConnection DSに付加する属性を示す図である。

【図15】一実施例の属性を含めたConnection DSのDDL定義を示す図である。

【図16】一実施例のルート・エレメントに付加する属性を示す図である。

【図17】一実施例の属性を含めたルート・エレメントのDDL定義を示す図である。

【図18】一実施例の属性を含めたルート・エレメントのDDL定義を示す図である。

【図19】一実施例の属性を含めたルート・エレメントのDDL定義を示す図である。

【図20】一実施例を用いたインターネットを介してのビデオ・コンテンツ検索のネットワークを示す図である。

【図21】インターネットを介したビデオ・コンテンツ検索の処理の流れを示すフローチャートである。

【図22】一実施例のMPEG-7記述伝送時の優先度を利用したストリーム構成方法を示す図である。

【図23】一実施例における属性useを含めたConnection DSに付加する属性の例である。

【図24】一実施例における属性useを含めたConnection DSのDDL定義である。

【図25】一実施例における参照宣言を導入する場合のReference DのDDL定義である。

【図26】一実施例における参照宣言を導入する場合のConnection DSとReference Declaration DSとに共通の属性のDDL定義である。

【図27】一実施例における参照宣言を導入する場合のConnection DSのDDL定義である。

【図28】一実施例におけるReference Declaration DSのDDL定義である。

【図29】一実施例における参照宣言を利用した接続先の参照方法を説明する図である。

【図1】

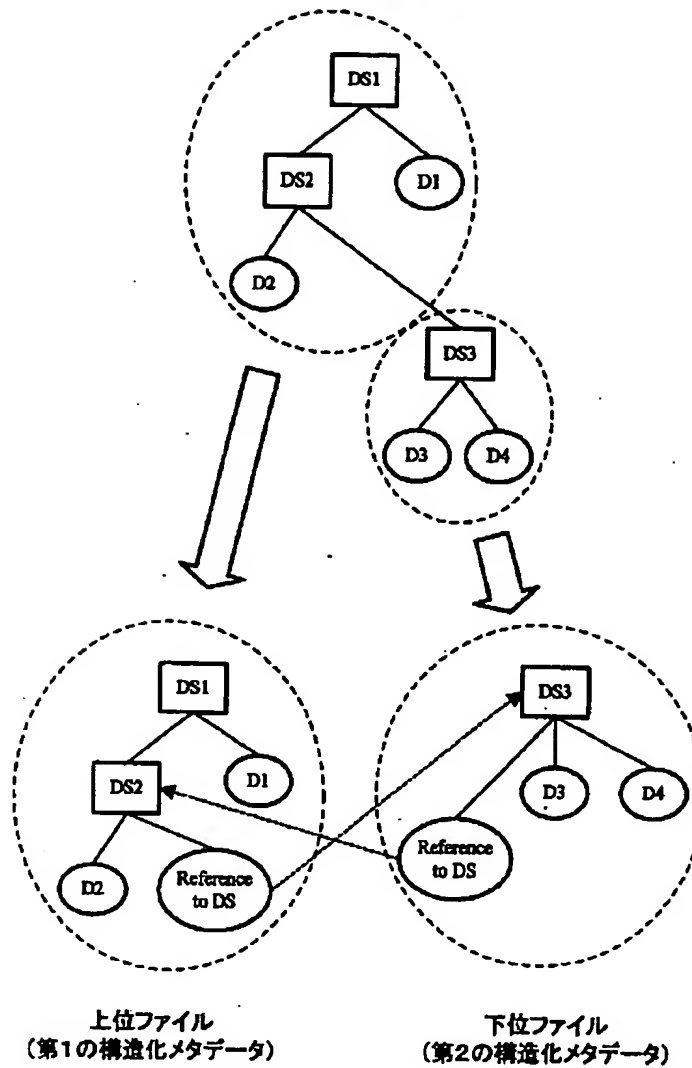


図1

【図12】

```

<!-- ***** -->
<!-- * Definition of the Temporary DS * -->
<!-- ***** -->

<complexType name="Temporary">
  <element name="Connection" type="mpeg7:Connection"
    minOccurs="1" maxOccurs="1"/>
  <any namespace="##any" minOccurs="1" maxOccurs="unbounded"
    processContents="skip"/>
</complexType>

```

図12

【図2】

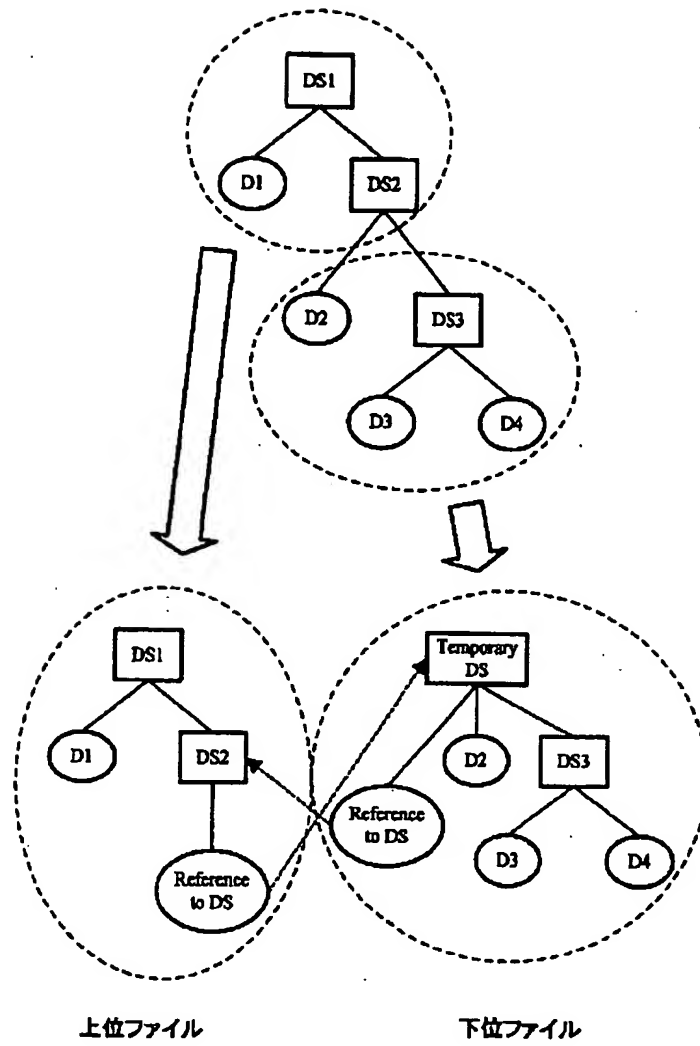


図2

【図3】

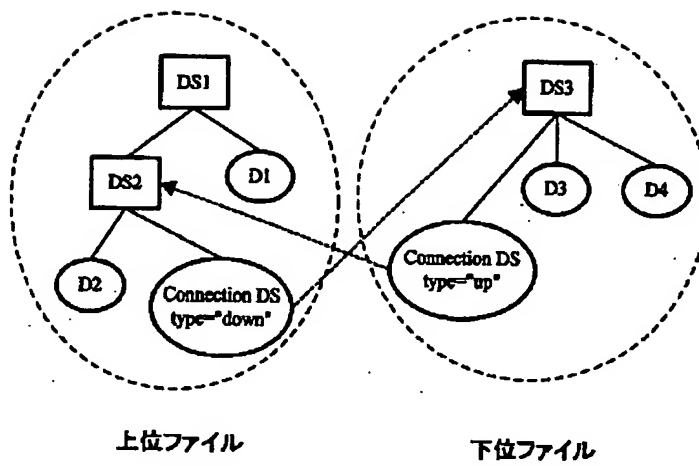
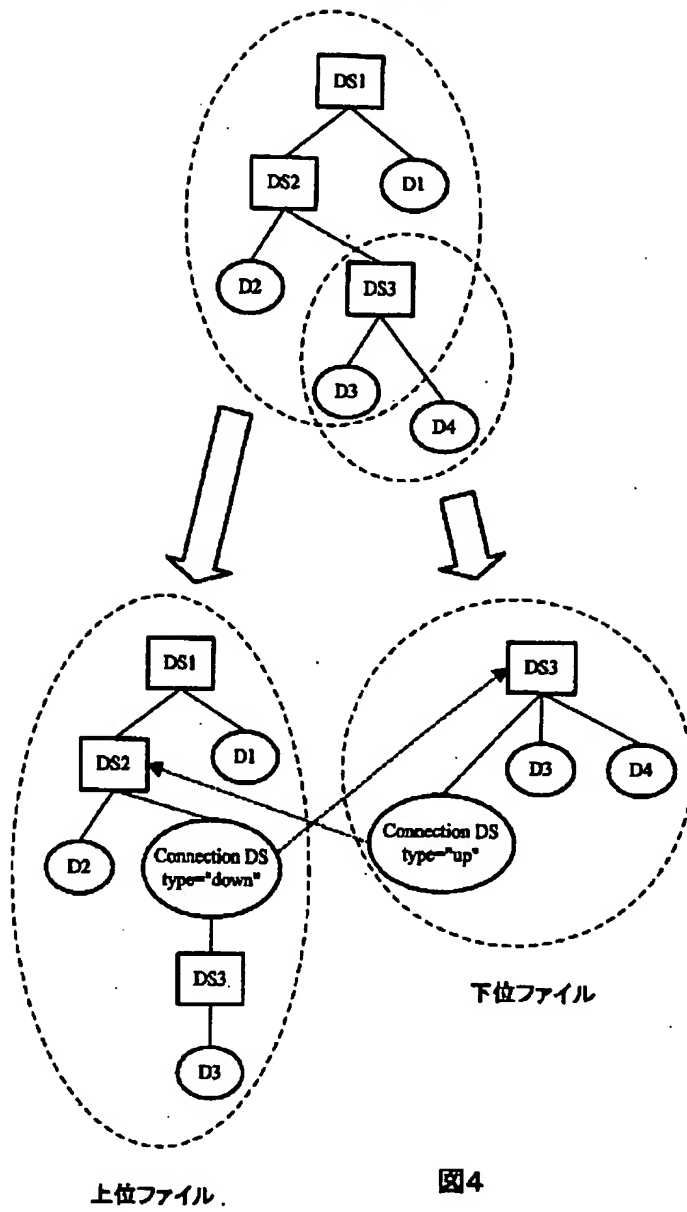


図3

【図4】



【図5】

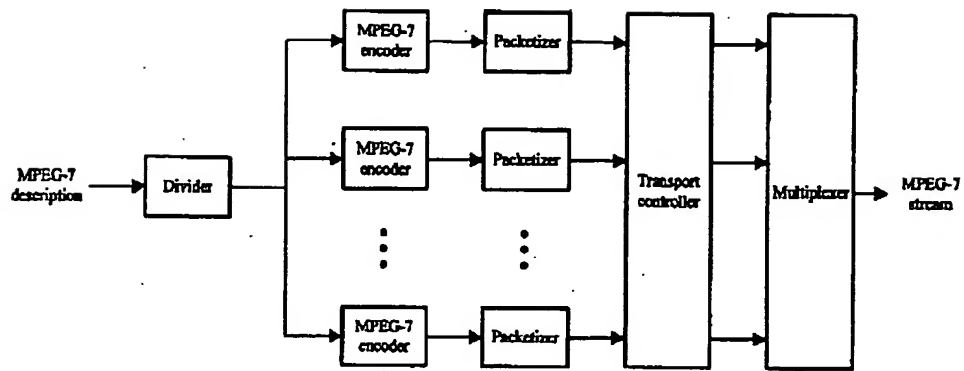


図6

【図6】

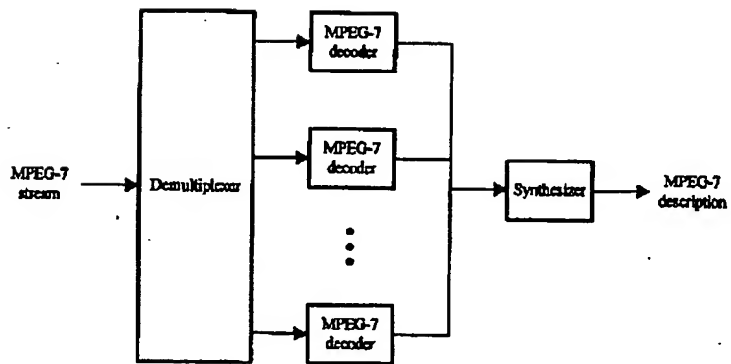


図6

【図7】

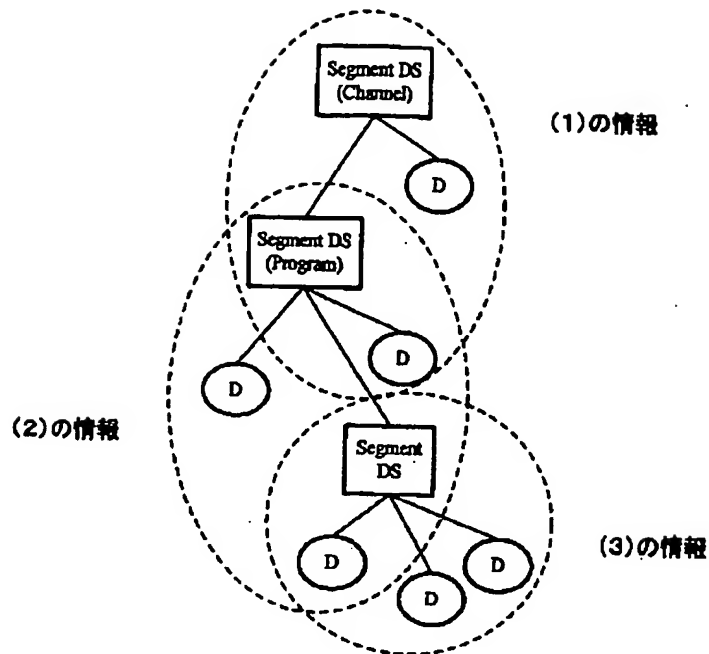


図7

【図11】

```

<!-- ***** -->
<!-- * Definition of the Connection DS * -->
<!-- ***** -->

<complexType name="Connection">
  <element name="Reference" type="mpeg7:Reference"
    minOccurs="1" maxOccurs="unbounded"/>
  <any namespace="##any" minOccurs="0" maxOccurs="unbounded"
    processContents="skip"/>
  <attribute name="type" use="required">
    <simpleType base="string">
      <enumeration value="up"/>
      <enumeration value="down"/>
    </simpleType>
  </attribute>
</complexType>

```

図11

【図8】

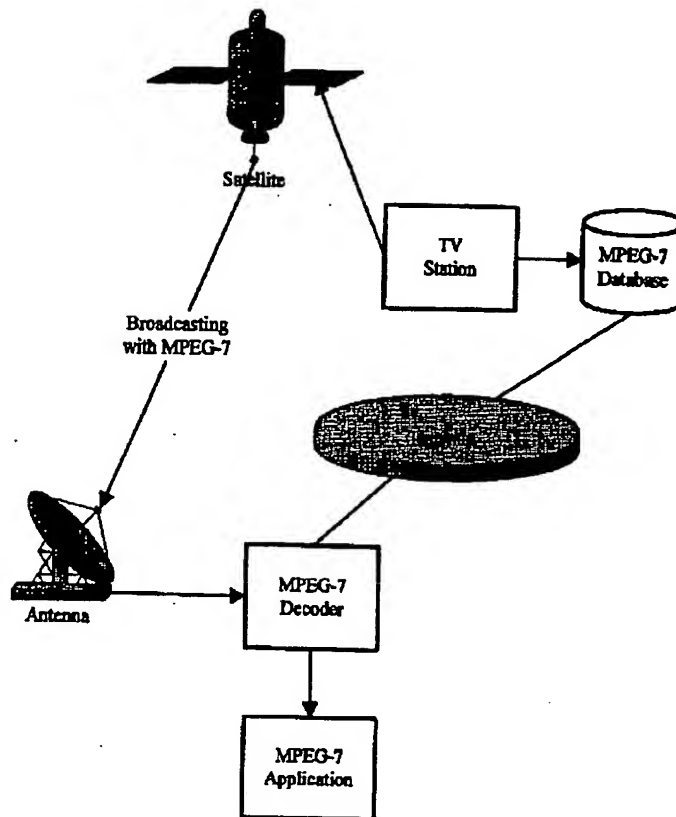


図8

【図17】

図17

```

<!-- ##### -->
<!-- Definition of the Mpeg7Root -->
<!-- ##### -->

<complexType name="Mpeg7RootType" abstract="true" content="elementOnly">
  <element name="ReferenceDeclaration"
    type="mpeg7:ReferenceDeclarationType" minOccurs="0"
    maxOccurs="1"/>
  <attribute name="id" type="ID" use="optional"/>
  <attribute name="version" type="string" use="optional"/>
  <attribute name="length" type="nonNegativeInteger" use="optional"/>
  <attribute name="priority" type="nonNegativeInteger" use="optional"/>
</complexType>

```

【図9】

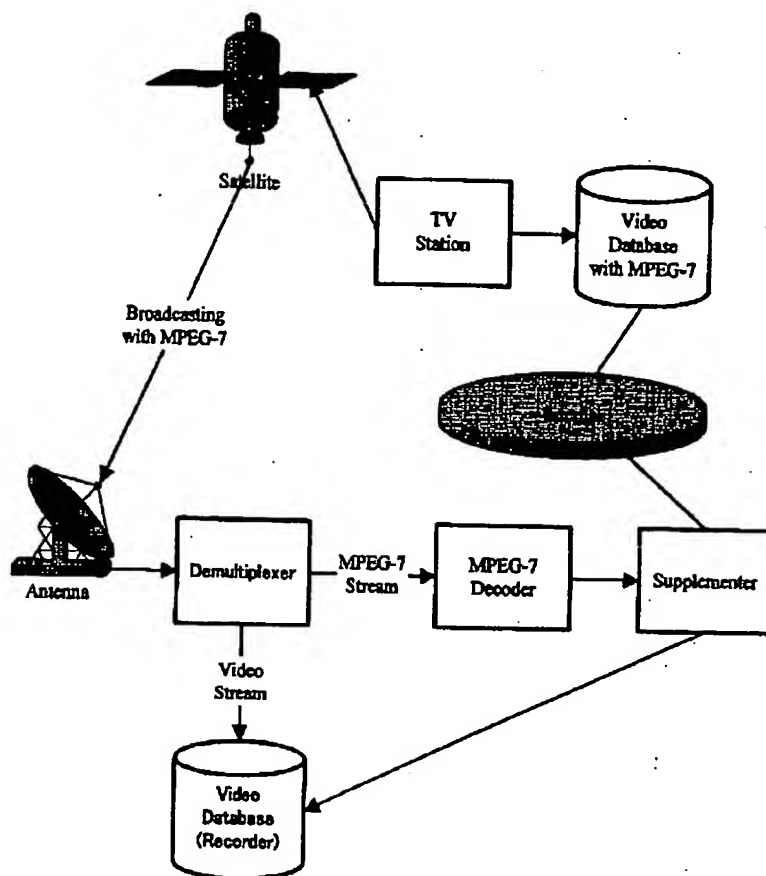


図9

【図25】

図25

```

<!-- ##### -->
<!-- Definition of the Reference D -->
<!-- ##### -->

<element name="Reference" type="mpeg7:ReferenceType"/>
<complexType name="ReferenceType" content="empty">
  <attribute name="href" type="uriReference" use="optional"/>
  <attribute name="idref" type="IDREF" use="optional"/>
  <attribute name="type" use="default" value="internal">
    <simpleType base="string">
      <enumeration value="internal"/>
      <enumeration value="external"/>
    </simpleType>
  </attribute>
</complexType>

```

【図10】

上位ファイル

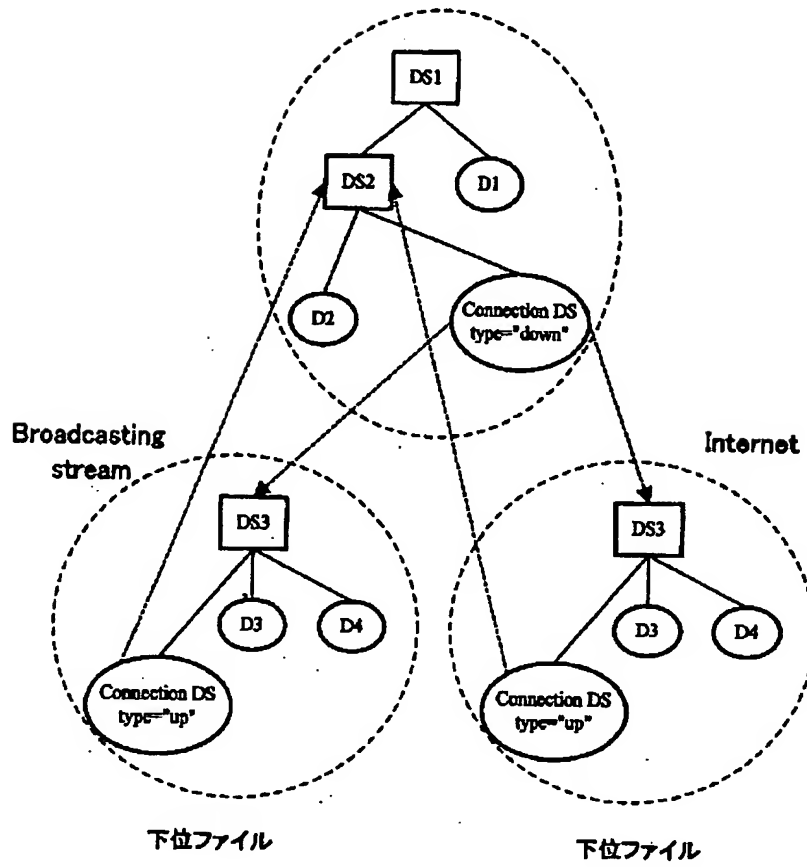


図10

【図13】

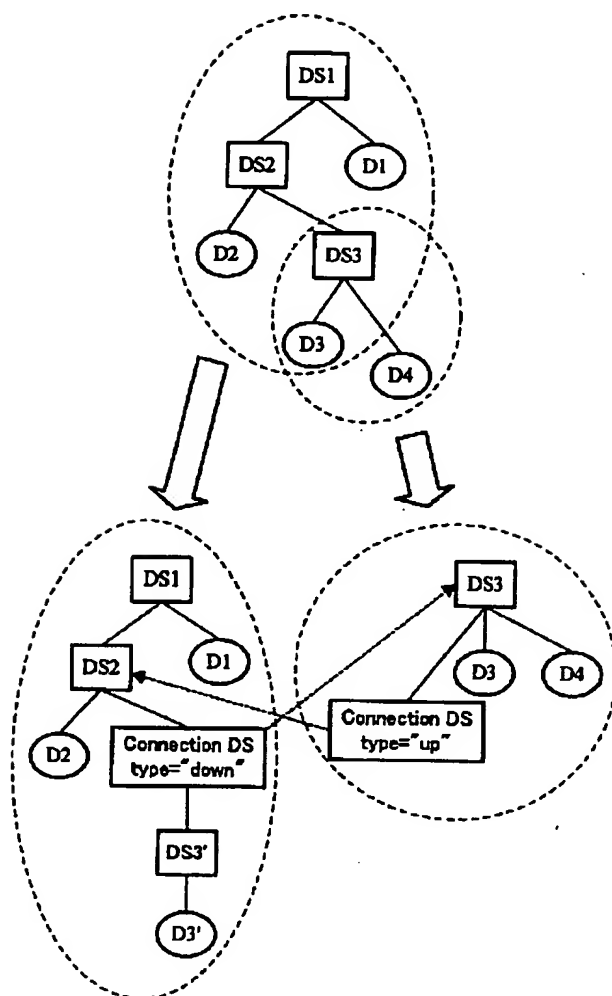


図13

【図14】

図14

属性名	型	意味
rootType	enumeration { main, unit }	参照先ファイルのルート・エレメントの種類
content	enumeration { media, contentCreationAndProduction, contentUsage, structuralAspects, conceptualAspects, contentNavigationAndAccess, contentOrganization, user, color, texture, shape, motion, localization, face, audioDescription, spokenContent, timbre, audioIndependent, header other }	参照先ファイルの内容の概略

【図15】

図15

```

<!-- ##### -->
<!-- Definition of the Connection DS -->
<!-- ##### -->

<element name="Connection" type="mpeg7:ConnectionType">
<complexType name="ConnectionType">
  <element name="Reference" type="mpeg7:ReferenceType" minOccurs="1"
    maxOccurs="unbounded"/>
  <any namespace="##any" minOccurs="0" maxOccurs="unbounded"
    processContents="skip"/>
  <attribute name="type" use="required">
    <simpleType base="string">
      <enumeration value="up"/>
      <enumeration value="down"/>
    </simpleType>
  </attribute>
  <attribute name="rootType">
    <simpleType base="string">
      <enumeration value="main"/>
      <enumeration value="unit"/>
    </simpleType>
  </attribute>
  <attribute name="content">
    <simpleType base="string">
      <enumeration value="media"/>
      <enumeration value="contentCreationAndProduction"/>
      <enumeration value="contentUsage"/>
      <enumeration value="structuralAspects"/>
      <enumeration value="conceptualAspects"/>
      <enumeration value="contentNavigationAndAccess"/>
      <enumeration value="contentOrganization"/>
      <enumeration value="user"/>
      <enumeration value="color"/>
      <enumeration value="texture"/>
      <enumeration value="shape"/>
      <enumeration value="motion"/>
      <enumeration value="localization"/>
      <enumeration value="face"/>
      <enumeration value="audioDescription"/>
      <enumeration value="spokenContent"/>
      <enumeration value="timbre"/>
      <enumeration value="audioIndependent"/>
      <enumeration value="header"/>
      <enumeration value="other"/>
    </simpleType>
  </attribute>
</complexType>

```

【図16】

図16

属性名	型	意味
length	nonNegativeInteger	ファイルの大きさ
priority	nonNegativeInteger	ファイルの優先度
content	enumeration { media, contentCreationAndProduction, contentUsage, structuralAspects, conceptualAspects, contentNavigationAndAccess, contentOrganization, user, color, texture, shape, motion, localization, face, audioDescription, spokenContent, timbre, audioIndependent, header other }	ファイルの内容の概略

【図18】

図18

```

<!-- ##### -->
<!-- Definition of the Mpeg7Main -->
<!-- ##### -->

<element name="Mpeg7Main" type="mpeg7:Mpeg7MainType"
    equivClass="MPEG7RootType"/>
<complexType name="Mpeg7MainType" base="mpeg7:Mpeg7RootType"
    derivedBy="extension">
    <choice minOccurs="1" maxOccurs="1">
        <element ref="ImageDescription"/>
        <element ref="VideoDescription"/>
        <element ref="AudioDescription"/>
        <element ref="AudioVisualDescription"/>
        <element ref="CollectionDescription"/>
        <element ref="UserDescription"/>
        <element ref="WorldDescription"/>
    </choice>
</complexType>

```

【図27】

図27

```

<!-- ##### -->
<!-- Definition of the Connection DS -->
<!-- ##### -->

<element name="Connection" type="mpeg7:ConnectionType">
<complexType name="ConnectionType">
    <element name="Reference" type="mpeg7:ReferenceType" minOccurs="1"
        maxOccurs="unbounded"/>
    <any namespace="##any" minOccurs="0" maxOccurs="unbounded"
        processContents="skip"/>
    <attribute name="type" use="required">
        <simpleType base="string">
            <enumeration value="up"/>
            <enumeration value="down"/>
        </simpleType>
    </attribute>
    <attribute name="rootType" type="mpeg7:referenceRootTypeType"
        use="optional"/>
    <attribute name="content" type="mpeg7:referenceContentType"
        use="optional"/>
    <attribute name="use" type="mpeg7:referenceUseType" use="default"
        value="optional"/>
</complexType>

```

【図19】

図19

```

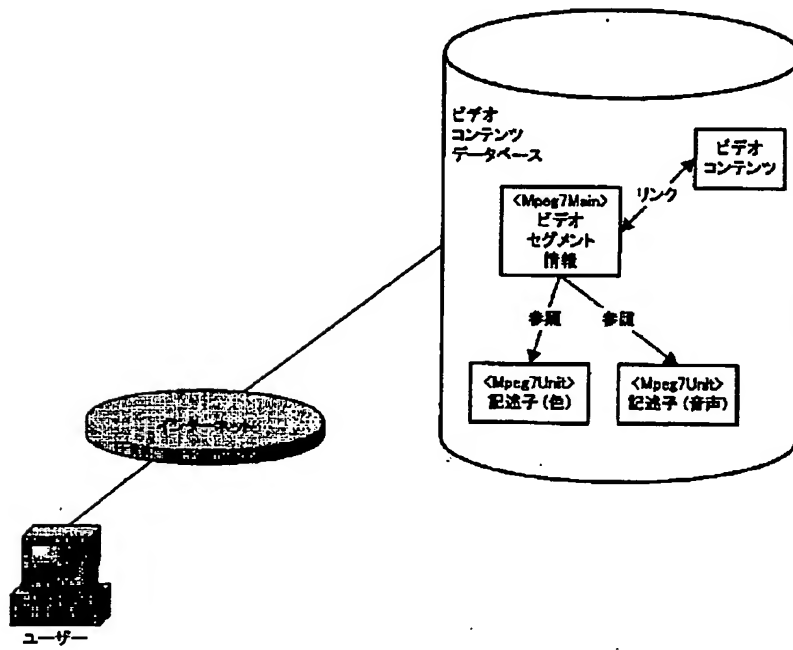
<!-- ##### -->
<!-- Definition of the Mpeg7Unit -->
<!-- ##### -->
<element name="Mpeg7Unit" type="mpeg7:Mpeg7UnitType"
    equivClass="mpeg7:Mpeg7RootType"/>
<complexType name="Mpeg7UnitType" base="mpeg7:Mpeg7RootType"
    derivedBy="extension">
    <any namespace="http://www.mpeg7.org" minOccurs="1" maxOccurs="1"
        processContents="lax"/>
    <attribute name="content" use="optional">
        <simpleType base="string">
            <enumeration value="media"/>
            <enumeration value="contentCreationAndProduction"/>
            <enumeration value="contentUsage"/>
            <enumeration value="structuralAspects"/>
            <enumeration value="conceptualAspects"/>
            <enumeration value="contentNavigationAndAccess"/>
            <enumeration value="contentOrganization"/>
            <enumeration value="user"/>
            <enumeration value="color"/>
            <enumeration value="texture"/>
            <enumeration value="shape"/>
            <enumeration value="motion"/>
            <enumeration value="localization"/>
            <enumeration value="face"/>
            <enumeration value="audioDescription"/>
            <enumeration value="spokenContent"/>
            <enumeration value="timbre"/>
            <enumeration value="audioIndependent"/>
            <enumeration value="header"/>
            <enumeration value="other"/>
        </simpleType>
    </attribute>
</complexType>

```

【図20】

図20

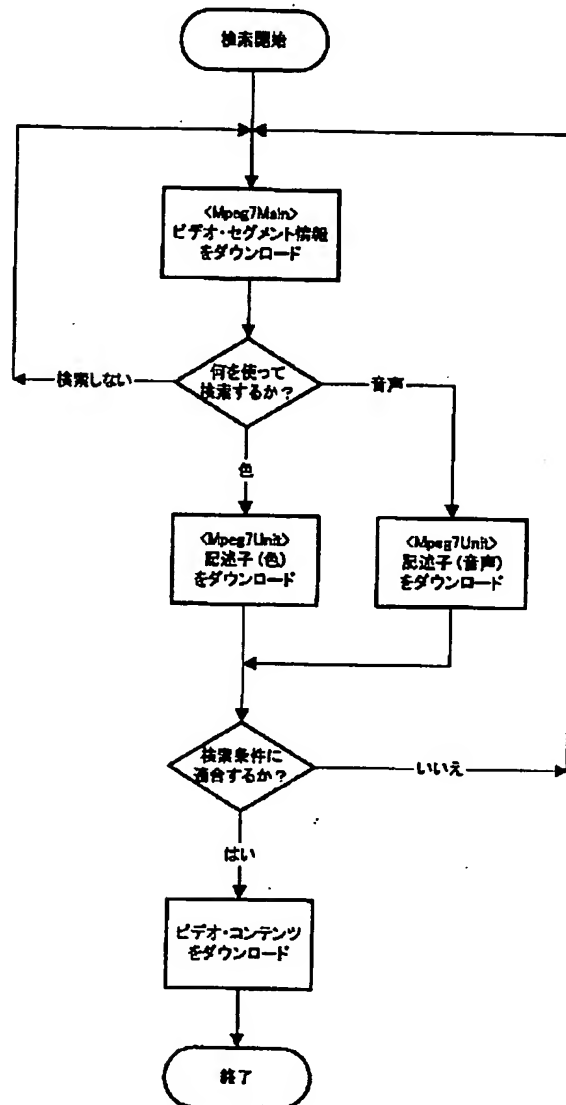
インターネットを介したビデオ・コンテンツ検索のネットワーク図



【図21】

図21

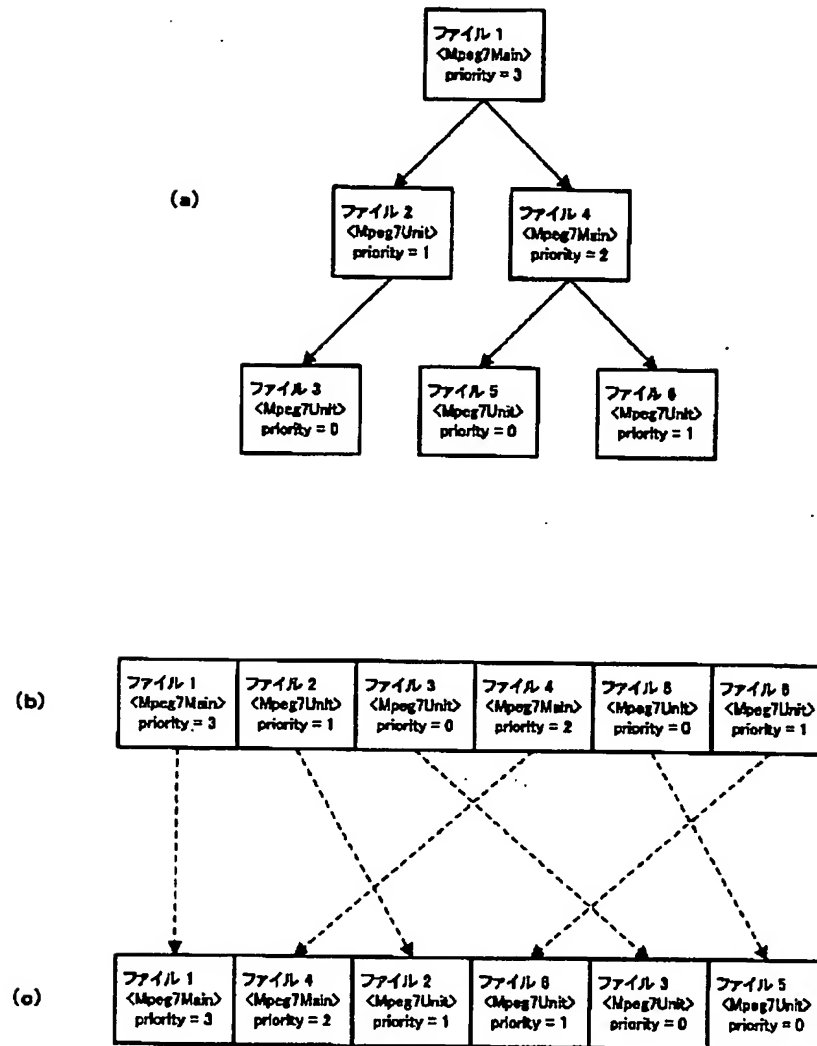
インターネットを介したビデオ・コンテンツ検索のフローチャート



【図22】

図22

MPEG-7記述伝送時の優先度を利用したストリーム構成方法



【図 23】

図 23

属性名	型	意味
rootType	enumeration { main, unit }	参照先ファイルのルート・エレメントの種類
content	enumeration { media, contentCreationAndProduction, contentUsage, structuralAspects, conceptualAspects, contentNavigationAndAccess, contentOrganization, user, color, texture, shape, motion, localization, face, audioDescription, spokenContent, timbre, audioIndependent, header other }	参照先ファイルの内容の概略
use	enumeration { required, optional }	参照先ファイルの必要性

【図24】

図24

```

<!-- ##### -->
<!-- Definition of the Connection DS -->
<!-- ##### -->

<element name="Connection" type="mpeg7:ConnectionType">
  <complexType name="ConnectionType">
    <element name="Reference" type="mpeg7:ReferenceType" minOccurs="1"
      maxOccurs="unbounded"/>
    <any namespace="##any" minOccurs="0" maxOccurs="unbounded"
      processContents="skip"/>
    <attribute name="type" use="required">
      <simpleType base="string">
        <enumeration value="up"/>
        <enumeration value="down"/>
      </simpleType>
    </attribute>
    <attribute name="rootType" use="optional">
      <simpleType base="string">
        <enumeration value="main"/>
        <enumeration value="unit"/>
      </simpleType>
    </attribute>
    <attribute name="content" use="optional">
      <simpleType base="string">
        <enumeration value="media"/>
        <enumeration value="contentCreationAndProduction"/>
        <enumeration value="contentUsage"/>
        <enumeration value="structuralAspects"/>
        <enumeration value="conceptualAspects"/>
        <enumeration value="contentNavigationAndAccess"/>
        <enumeration value="contentOrganization"/>
        <enumeration value="user"/>
        <enumeration value="color"/>
        <enumeration value="texture"/>
        <enumeration value="shape"/>
        <enumeration value="motion"/>
        <enumeration value="localization"/>
        <enumeration value="face"/>
        <enumeration value="audioDescription"/>
        <enumeration value="spokenContent"/>
        <enumeration value="timbre"/>
        <enumeration value="audioIndependent"/>
        <enumeration value="header"/>
        <enumeration value="other"/>
      </simpleType>
    </attribute>
    <attribute name="use" use="default" value="optional">
      <simpleType base="string">
        <enumeration value="required"/>
        <enumeration value="optional"/>
      </simpleType>
    </attribute>
  </complexType>

```

【図26】

図26

```

<!-- ##### -->
<!-- Attributes of the Connection DS -->
<!-- ##### -->

<sympleType name="referenceRootTypeType" base="string">
    <enumeration value="main"/>
    <enumeration value="unit"/>
</sympleType>

<sympleType name="referenceContentType" base="string">
    <enumeration value="media"/>
    <enumeration value="contentCreationAndProduction"/>
    <enumeration value="contentUsage"/>
    <enumeration value="structuralAspects"/>
    <enumeration value="conceptualAspects"/>
    <enumeration value="contentNavigationAndAccess"/>
    <enumeration value="contentOrganization"/>
    <enumeration value="user"/>
    <enumeration value="color"/>
    <enumeration value="texture"/>
    <enumeration value="shape"/>
    <enumeration value="motion"/>
    <enumeration value="localization"/>
    <enumeration value="face"/>
    <enumeration value="audioDescription"/>
    <enumeration value="spokenContent"/>
    <enumeration value="timbre"/>
    <enumeration value="audioIndependent"/>
    <enumeration value="header"/>
    <enumeration value="other"/>
</sympleType>

<sympleType name="referenceUseType" base="string">
    <enumeration value="required"/>
    <enumeration value="optional"/>
</sympleType>

```

【図28】

図28

```

<!-- ##### -->
<!-- Definition of the Reference Declaration DS -->
<!-- ##### -->

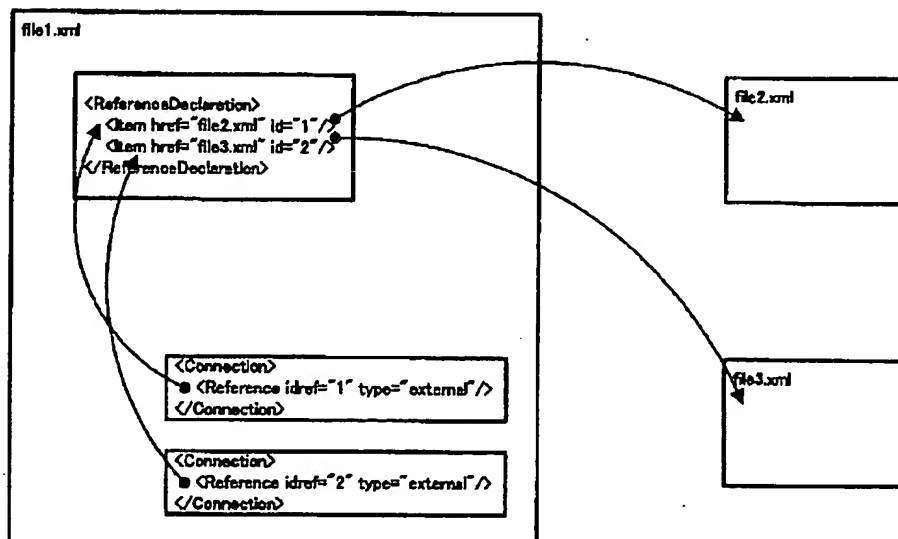
<element name="ReferenceDeclaration" type="mpeg7:ReferenceDeclarationType"/>
<complexType name="ReferenceDeclarationType">
  <element name="Item" minOccurs="1" maxOccurs="unbounded"/>
  <complexType content="empty">
    <attribute name="href" type="uriReference"
      use="required"/>
    <attribute name="id" type="ID" use="required"/>
    <attribute name="rootType"
      type="mpeg7:referenceRootTypeType"
      use="optional"/>
    <attribute name="content"
      type="mpeg7:referenceContentType"
      use="optional"/>
    <attribute name="use" type="mpeg7:referenceUseType"
      use="default" value="optional"/>
  </complexType>
</element>
</complexType>

```

【図29】

図29

参照宣言を利用した接続先の参照方法を説明する図



フロントページの続き

(31)優先権主張番号 特願2000-277796(P2000-277796)
(32)優先日 平成12年9月13日(2000. 9. 13)
(33)優先権主張国 日本(JP)

(31)優先権主張番号 特願2000-288180(P2000-288180)
(32)優先日 平成12年9月22日(2000. 9. 22)
(33)優先権主張国 日本(JP)